CPSC 1155 – Lab 5

Selection Statements

Learning Objectives

At the end of this lab, you should be able to:

- Understand and analyze a selection statement
- Write a pseudocode with selection statements to solve a problem
- Write and run a C++ program with selection statements to solve a problem

Lab Introduction

This lab focuses on learning selection statements.

Lab Readings

1. Pseudocode Lecture
2. Chapter 3 – Selection

Lab Instructions

Write your answers to the **Practice Questions** in a text editor (answer.txt).

For each **Problem Statement**, follow the steps below:

1. Read the problem statement and clarify the problem.
    a. Break the problem into smaller problems if needed.
2. Determine the IPO.
    a. Determine input, output, **intermediate variables**, constants, and conditions.
    b. Declare the variables and constants (data type + meaningful names).
    c. Work out the problem by hand using typical input values. Determine the range of valid input values.
    d. Determine the process.
3. Write a pseudocode as required.
4. Write a C++ program (use the given filename) that implements the pseudocode.
    a. Add comments where needed. Make sure to use a comments header to reflect the intention of your program and name of the author (you) and the date the program was written.
    b. Test, debug, and execute the program using typical values.

Submit according to the instruction in the "Lab Submission" section.

Practice Questions

1. [5] Suppose x and y are two integers. Give Boolean expressions, using the && and || operators, for the following tests. Don't use the Boolean ! operator. But you can use != to check if a number is not equal to zero.
    a. Both x and y are zero

b. At least one of them is zero

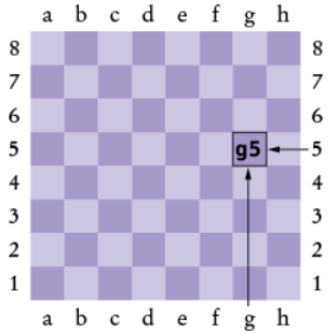
c. At most one of them is zero

d. Neither of them is zero

e. Exactly one of them is zero

2. [4] Rush hour is between 7:00 and 9:00 or 16:00 and 18:00 (in military time). Draw the flow chart that correctly sets the rush_hour variable to "Y" during rush hour and to "N" otherwise, when the hour is stored in the variable hr.

3. [4] Each square on a chess board can be described by a letter and number, such as g5 in this picture. Write nested if statements to determine the color of a square based on the letter and number. You can assume letter is stored in variable L and number is stored in variable N.



4. [3] A year with 366 days is called a leap year. Leap years are necessary to keep the calendar synchronized with the sun because the earth revolves around the sun once every 365.25 days. Actually, that figure is not entirely precise, and for all dates after 1582 the Gregorian correction applies. Usually years that are divisible by 4 are leap years, for example 1996. However, years that are divisible by 100 (for example, 1900) are not leap years, but years that are divisible by 400 are leap years (for example, 2000). Write a single if statement with Boolean operators to determine if a year is a leap year.

5. [6] Your C++ program is supposed to prompt for time (hour) and print "night" if time is before 6, "morning" if time is before 12, "afternoon" if time is before 18, and "evening" otherwise. Test the following programs with at least four test cases (different hours) for each version and explain why they don't work correctly. Write the correct selection statements for this program.

Version 1	Version 2

`cin >> hour;`
`if (hour < 6)`
`  cout << "night";`
`if (hour < 12)`
`  cout << "morning";`
`if (hour < 18)`
`  cout << "afternoon";`
`if (hour < 24)`
`  cout << "evening";`
	`cin >> hour;`
	`if (hour < 18)`
	`  cout << "afternoon";`
	`else if (hour < 12)`
	`  cout << "morning";`
	`else if (hour < 6)`
	`  cout << "night";`
	`else`
	`  cout << "evening";`

Problem Statements

6. [6] (checkNumbers.cpp) Write a **C++ program** that prompts the user to enter two integers and compares them.

If the numbers are invalid (smaller than 1), the program prints error message and exit.

If the numbers are equal, the program prints that the numbers are equal and ends.

If the numbers are not equal, the program prints the larger number.

Then, the program checks whether the larger number is divisible by the other one and displays a message accordingly (You must use nested if).

Here are sample runs:

```
Enter two integers: 2 3
3 is larger than 2
3 is not divisible by 2

Enter two integers: 4 2
4 is larger than 2
4 is divisible by 2

Enter two integers: 5 5
5 is equal to 5

Enter two integers: 0 9
Input must be greater than 0
```

Lab Submission

Submit a zip folder named as yourName_Lab5.zip to Brightspace. This folder should consist of a text file named **answer.txt** with your answers to Practice Questions and **checkNumbers.cpp** for Problem Statement 6.

Please make sure that all your .cpp files compile and run properly before submission. Your file must run properly in order to receive full marks.

Marking Scheme

The marks are given in square brackets [ ] for each question.